EXHIBIT 1

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/709,477	02/21/2007	Pradip Kulkarni	PNTA-P006.CIP.A	4127

70848 7590 01/05/2010

PANTA c/o MURABITO HAO & BARNES LLP
Two Market Street
3rd Floor
San Jose, CA 95113

EXAMINER	
MARTELLO, EDWARD	

ART UNIT	PAPER NUMBER
2628	

MAIL DATE	DELIVERY MODE
01/05/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	11/709,477	KULKARNI ET AL.	
	Examiner	Art Unit	
	Edward Martello	2628	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *21 February 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *21 February 2007* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Specification

1. The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required: Claims 4, 5, 14, 16 and 17 all incorporate features called "union block devices." There is insufficient basis within the claims to define the character of these devices. Union block devices are only described in paragraph [0024] of the instant application which has this single sentence: "System 200 has a number of compute nodes 220a-n coupled to first storage unit 240 and a corresponding second storage unit 250a-n though a corresponding union block device (UBD) 230a-n." This description is insufficient to define this device. The union block devices are also shown in figure 2 as functional blocks without any further details. There is insufficient information provided as to the functionality of this joining device and as such the specification provides insufficient antecedent basis for the claimed subject matter.

2. Claims 5 and 7 proclaim that the union device consists of low-level drivers which are generally known in the art as computer processor instructions, usually running at the OS kernel level, that provide low-level hardware device interfacing to higher level software applications. Based on these claims, the union device must contain a processor or some dedicated hardware that can process driver instructions, adding to the vagueness of these devices.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

3. Claims 4, 5, 14, 16 and 17 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claims 4, 5, 14, 16 and 17 all incorporate features called "union block devices" with insufficient basis within the claims to define their character. Union block devices are only described in paragraph [0024] of the instant application which has this single sentence: "System 200 has a number of compute nodes 220a-n coupled to first storage unit 240 and a corresponding second storage unit 250a-n though a corresponding union block device (UBD) 230a-n." This description is insufficient to define this device. The union block devices are also shown in figure 2 as functional blocks without any further details. There is insufficient information provided in the claims and the specification as to the functionality of this joining device and as such, the claimed subject matter does not meet the requirements of UCS 112 second paragraph.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

4. Claims 23-27 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Each of the claims recites the feature or limitation of logic which is encoded in one or more tangible media. Abstract ideas or methods are examples of non-functional descriptive material which do not encompass any of the four statutory categories of invention. Rephrasing these claims to call out tangible media with or comprising computer

executable instructions that perform the logic of the methods described could result in statutory claims if done properly.

5. Claims 23-27 are rejected under 35 U.S.C. § 101 because the claimed invention is directed to non-statutory subject matter. In addition to the above described problem, the Applicants describe in paragraph [0023] of the specification that the computer readable or tangible media may be any medium capable or carrying instructions and includes modulated data signals. Thus, in light of the specification, claims 23-27 recite a signal per se. Claims that recite nothing but the physical characteristics of a form of energy, such as a frequency, voltage, or the strength of a magnetic field, define energy or magnetism, per se, and as such are nonstatutory natural phenomena. O'Reilly, 56 U .S. (15 How.) at 112-14.

6. Moreover, it does not appear that a claim reciting a signal encoded with "computer readable codes" falls within any of the categories of patentable subject matter set forth in 35 U.S.C. §101. The codes must be in a form that will cause a computer or CPU or processor to execute a method as presented in the respective claims.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

7. Claims 14-27 are rejected under 35 U.S.C. 102(b) as being anticipated by Chang (U. S. Patent Application Publication 2003/0126242 A1, hereafter '242).

8. In regard to claim 14, Chang teaches a system for indexing file systems for a plurality of compute nodes (‘242; fig. 1), comprising: a first storage unit for storing blocks of a root image of said compute nodes (‘242; fig. 1, element 72); a plurality of second storage units for storing leaf images of respective compute nodes (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific blocks are preferably allocated and/or updated when writes or similar commands or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)), said leaf images comprising additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific blocks are preferably allocated and/or updated when writes or similar commands or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)); and a plurality of union block devices corresponding to said compute nodes (‘242; ¶ 0020, iSCSI drivers and protocol in element 60 of fig. 1), said union block devices for interfacing between said compute nodes and said first and second storage units to distribute said file systems to said compute nodes, wherein said union block devices create said file systems by merging the blocks of said root image stored on the first storage unit with the blocks of respective leaf images stored on respective second storage unit (‘242; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients), and does not explicitly teach wherein further at least one of said compute nodes is configurable to index said root image and provide the indexing results to another of said compute nodes. a plurality of union block devices corresponding to said compute nodes (‘576; ¶ 0020, iSCSI drivers and protocol in element 60 of fig. 1), said union

block devices for interfacing between said compute nodes and said first and second storage units to distribute said file systems to said compute nodes (‘576; ¶ 0020, iSCSI drivers and protocol in element 60 of fig. 1), wherein said union block devices create said file systems by merging the blocks of said root image stored on the first storage unit with the blocks of respective leaf images stored on respective second storage units (‘576; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients), wherein further at least one of said compute nodes is configurable to index (‘242; ¶ 0022-0026, snapshot methods) said root image and provide the indexing results to another of said compute nodes (‘242; ¶ 0022-0026, snapshot methods).

9. Regarding claim 15, Chang further teaches wherein said first storage unit and said second storage units are contained within a single storage appliance (‘242; fig. 1, element 60; ¶ 0017, storage controller or server).

10. In regard to claim 16, Chang teaches the system as recited in Claim 14 and further teaches the system as further comprising: a plurality of union block devices for interfacing between respective compute nodes (‘242; ¶ 0020, iSCSI drivers and protocol in element 60 of fig. 1) and said first storage unit and respective second storage units (‘242; fig. 1, element 60; ¶ 0017, storage controller or server), said union block devices operable to distribute application environments to the compute nodes, wherein said union block devices create said application environments by merging the blocks of said root image with the blocks of respective leaf images (‘242; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients).

11. Regarding claim 17, Chang further teaches wherein said union block devices comprise low-level drivers for interfacing between the file systems of respective compute nodes and said first storage unit, respective second storage units, and said cache (‘242; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients).

12. In regard to claim 18 Chang teaches the system as recited in Claim 14 and further teaches wherein said first storage unit is read-only (‘242; ¶ 0012, writes do not touch common OS and application image blocks).

13. Regarding claim 19, Chang teaches a method for indexing file systems for a plurality of compute nodes (‘242; fig. 1), comprising: storing blocks of a root image of said compute nodes on a first storage unit (‘242; fig. 1, element 72); storing leaf images for respective compute nodes on respective second storage units (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific blocks are preferably allocated and/or updated when writes or similar commands or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)), said leaf images comprising additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of the root image (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific blocks are preferably allocated and/or updated when writes or similar commands or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)); merging the blocks of said root image with the blocks of respective leaf images stored on respective second storage units to create respective file systems for respective compute nodes (‘242 ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client

images in element 70 of fig. 1 to be distributed to clients); receiving indexing results pertaining to said root image from one of said compute nodes (‘242; ¶ 0022-0026, snapshot methods); and providing said indexing results to the others of said compute nodes (‘242; ¶ 0022-0026, snapshot methods).

14. In regard to claim 20, Chang teaches the method as recited in Claim 19 and further teaches the method as further comprising: storing said indexing results on a shared storage unit (‘242; fig. 1, element 60; ¶ 0017, storage controller or server).

15. Regarding claim 21, Chang teaches the method as recited in Claim 19 and further teaches wherein said merging occurs at an operational level between the respective file systems of the compute nodes and said first storage unit and respective second storage units (‘242; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients).

16. In regard to claim 22, Chang teaches the method as recited in Claim 19 and further teaches wherein said first storage unit is read-only (‘242; ¶ 0012, writes do not touch common OS and application image blocks).

17. Regarding claim 23 Chang teaches logic (methods) encoded in one or more tangible media for execution by a first compute node, and when executed said logic operable to (‘242; fig. 1): receive data blocks of a file system, said data blocks comprising a root image portion (‘242; fig. 1, element 72) and leaf image portion, said leaf image portion comprising additional data blocks not previously contained in said root image portion and changes made by said first compute node to the blocks of said root image (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific blocks are preferably allocated and/or updated when writes or similar commands

or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)), wherein said file system is the result of merging said root image portion and said leaf image portion together at the block-level (‘242; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients); index said root image portion (‘242; ¶ 0022-0026, snapshot methods); and provide the results of said indexing to a second compute node (‘242; ¶ 0022-0026, snapshot methods).

18. In regard to claim 24, Chang further teaches wherein said providing further comprises: storing said results of said indexing on a shared storage unit accessible by said second compute node (‘242; ¶ 0022-0026, snapshot methods).

19. Regarding claim 25 Chang teaches the logic (methods) as recited in Claim 23 and further teaches wherein said logic is further operable to: index said leaf image portion (‘242; ¶ 0022-0026, snapshot methods).

20. In regard to claim 26, Chang further teaches wherein said logic is further operable to: re-index said file system by re-indexing said leaf image portion and merging the results of said re-indexing of said leaf image portion with said results of said indexing of said root image portion (‘242; ¶ 0022-0026, snapshot methods capture updates to client specific images).

21. Regarding claim 27, Chang teaches the logic as recited in Claim 23 and further teaches wherein said root image portion is read-only (‘242; ¶ 0012, writes do not touch common OS and application image blocks).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

22. Claims 1-13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chang (U. S. Patent Application Publication 2003/0126242 A1, hereafter ‘242) as applied to claims 14-27 above and in view of Kobayashi et al. (U. S. Patent 6,101,576, hereafter ‘576).

23. Regarding claim 1, Chang teaches a system for providing data to a plurality of compute nodes (‘242; fig. 1) comprising: a first storage unit for storing blocks of a root image of said compute nodes (‘242; fig. 1, element 72); a plurality of second storage units for storing leaf images of respective compute nodes (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific blocks are preferably allocated and/or updated when writes or similar commands or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)), said leaf images comprising additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of

said root image nodes (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific blocks are preferably allocated and/or updated when writes or similar commands or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)); and does not explicitly teach a physical cache for caching blocks of said root image previously accessed by at least one of said compute nodes. Chang does teach caching blocks of said root image previously accessed by at least one of said compute nodes (‘242; ¶ 0010; caching of original blocks is provided in the original volume) and Kobayashi, solving the same problem of caching frequently accessed data sets, however, explicitly teaches physical caches (‘576; fig. 12, memory caches as elements 11a, 11b and a disk cache as element 14b; col. 5, ln. 47-51) for the benefit of increasing the speed of data transfers from the host to client in data streaming or transfer operations as frequently accessed data is kept in high speed memory and not constantly retrieved from slower disk sources. It would have been obvious to one of ordinary skill in the art at the time of the invention to have combined the detailed image/data cache teachings of Kobayashi with the server/client image assess and remotely stored boot/OS system teachings of Chang for the benefit of increasing the speed of data transfers from the host to client in data streaming or transfer operations as frequently accessed data is kept in high speed memory and not constantly retrieved from slower disk sources.

24. In regard to claim 2, Kobayashi further teaches wherein said cache is configurable to store the X most recently accessed blocks of said root image, and wherein further X represents a cache threshold value (‘576; col. 2, ln. 37 through col. 3, ln. 9).

25. Regarding claim 3, Chang and Kobayashi teach the system as recited in Claim 1 Chang further teaches wherein said first storage unit, said second storage units, and said cache are

contained within a single storage appliance (‘242; fig. 1, element 60; ¶ 0017, storage controller or server).

26. In regard to claim 4, Chang and Kobayashi teach the system as recited in Claim 1 and Chang further comprising: a plurality of union block devices for interfacing between respective compute nodes and said first storage unit, respective second storage units, and said cache (‘242; ¶ 0020, iSCSI drivers and protocol in element 60 of fig. 1) to distribute application environments to the compute nodes, wherein said union block devices create said application environments by merging the blocks of said root image with the blocks of respective leaf images (‘242; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients).

27. Regarding claim 5, Chang further teaches wherein said union block devices comprise low-level drivers for interfacing between the file systems of respective compute nodes and said first storage unit, respective second storage units, and said cache (‘242; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients).

28. In regard to claim 6, Chang and Kobayashi teach the system as recited in Claim 1 wherein said first storage unit is read-only (‘242; ¶ 0012, writes do not touch common OS and application image blocks).

29. Regarding claim 7, Chang teaches a method for providing data to a plurality of compute nodes(‘242; fig. 1), comprising: storing blocks of a root image of said compute nodes on a first storage unit (‘242; fig. 1, element 72); storing leaf images for respective compute nodes on respective second storage units (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific

blocks are preferably allocated and/or updated when writes or similar commands or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)), said leaf images comprising additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of the root image (‘242; fig. 1, elements 74, 76, 78; ¶ 0012, The client-specific blocks are preferably allocated and/or updated when writes or similar commands or messages are received (e.g., writes do not touch common OS and application image blocks but instead are client-specific)); and does not explicitly teach caching blocks of said root image that have been accessed by at least one of said compute nodes in a cache memory. Chang does teach caching blocks of said root image previously accessed by at least one of said compute nodes (‘242; ¶ 0010; caching of original blocks is provided in the original volume) and Kobayashi, solving the same problem of caching frequently accessed data sets, however, teaches explicit physical caches (‘576; fig. 12, memory caches as elements 11a, 11b and a disk cache as element 14b; col. 5, ln. 47-51) for the benefit of increasing the speed of data transfers from the host to client in data streaming or transfer operations as frequently accessed data is kept in high speed memory and not constantly retrieved from slower disk sources. It would have been obvious to one of ordinary skill in the art at the time of the invention to have combined the detailed image/data cache teachings of Kobayashi with the server/client image assess and remotely stored boot/OS system teachings of Chang for the benefit of increasing the speed of data transfers from the host to client in data streaming or transfer operations as frequently accessed data is kept in high speed memory and not constantly retrieved from slower disk sources.

30. In regard to claim 8, Chang further teaches the method as further comprising: receiving a read request from at least one of said compute nodes, wherein a first portion of the data requested is currently stored in said cache memory; and providing said first portion of said data to said at least one of said compute nodes from said cache memory (‘242; ¶ 0020-0022).

31. Regarding claim 9, Chang further teaches the method as further comprising: updating said cache memory based on said read request (‘242; ¶ 0020-0022).

32. In regard to claim 10, Kobayashi further teaches wherein a second portion of the data requested is not currently stored in said cache memory and said updating comprises: caching said second portion in said cache memory; and removing the least recently accessed data from said cache memory if the amount of data is said cache memory is above a threshold value (‘576; col. 12, ln. 59-65, LRU algorithm).

33. Regarding claim 11, Chang and Kobayashi teach the method as recited in Claim 7 and Chang further teaches the method as further comprising: merging the blocks of said root image with the blocks of respective leaf images to create cohesive respective application environments (‘242 ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients).

34. In regard to claim 12, Chang further teaches wherein said merging occurs at an operational level between the respective file systems of the compute nodes and said first storage unit, respective second storage units, and said cache memory (‘242; ¶ 0020-0021, iSCSI drivers and protocol in element 60 of fig. 1 to allow output of pooled base and client images in element 70 of fig. 1 to be distributed to clients).

35. Regarding claim 13, Chang and Kobayashi teach the method as recited in Claim 7 and Chang further teaches wherein said first storage unit is read-only (‘242; ¶ 0012, writes do not touch common OS and application image blocks).

Conclusion

The following prior art, made of record, was not relied upon but is considered pertinent to applicant's disclosure:

US 6751658 B1	Providing a reliable operating system for clients of a net-booted environment
US 6421777 B1	Method and apparatus for managing boot images in a distributed data processing system – Teaches read-only boot data and client specific images in a networked boot environment.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Edward Martello whose telephone number is (571) 270-1883. The examiner can normally be reached on M-F 7:30-5:00 EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner’s supervisor, Xiao Wu can be reached on (571) 272-7761. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/EM/

Examiner, Art Unit 2628

/XIAO M. WU/
Supervisory Patent Examiner, Art Unit 2628

REMARKS

Claims 1-27 are pending. Claims 5, 8-13, 15-18, 20-22, and 23-27 are amended. No new matter has been added as a result of these amendments.

Claim Objections

Claims 5, 8-13, 15-18, 20-22, and 24-27 are objected to because of typographical errors. Claims 5, 8-13, 15-18, 20-22, and 24-27 have been amended as appropriate.

Objection to the Specification

Paragraph 22 of the Specification is objected to as it calls out "dashed line 105," while the dashed line in Figure 1 is labeled "106." Paragraph 22 of the Specification has been amended as appropriate.

The Specification is further objected to as failing to provide proper antecedent basis for the claimed subject matter. In particular, the rejection contends that Claims 4, 5, 14, 16, and 17 all allegedly incorporate features called "union block devices" with insufficient basis within the claims to define their character. Applicants respectfully disagree. Applicants respectfully assert that Figure 2 illustrates an exemplary block-level distributed application environment that can contain both physical and virtual elements (Paragraph 26). As illustrated in Figure 2 and discussed in Paragraphs 12 and 22-29, a "union block device" is a driver implemented by and for each respective compute node. As such each union block device driver provides an interface between the first and second storage devices and the file system of its compute node, providing the compute

node with an instantiation of the application environment by accessing the compute node's leaf image containing portions of the application environment that the compute node has changed as well as locating portions of the application environment that have not been changed by the compute node (Paragraphs 27-28). In one exemplary embodiment, the portions that have not been changed by the compute node can be found in the root image (Paragraph 28). Applicants respectfully assert that one skilled in the art would understand the nature and function of such a union block device driver, as described in Paragraphs 12 and 22-29. Applicants respectfully assert that drivers serving as interfaces between operating system file systems for a computer system and associated storage devices are well known in the art. Further, one skilled in the art would understand how a driver may serve as an interface between storage devices and a file system of a computer system.

As understood by Applicants, the rejection contends that Figure 2 illustrates a union block device implemented as a stand-alone "device" that would need its own separate processor. Applicants respectfully disagree. As discussed above, Applicants respectfully assert that Figure 2 illustrates an exemplary block-level distributed application environment that can contain both physical and virtual elements (Paragraph 26). As illustrated in Figure 2 and discussed in Paragraphs 12 and 22-29, a "union block device" is therefore a driver implemented by and for a particular compute node. In other words, union block devices are not "devices" as understood by the rejection, requiring their own separate processor to run the driver. Rather, union block devices are drivers implemented on and for each compute node.

As understood by Applicants, the rejection further questions how a union block device can create a new leaf image on a respective second storage union when the respective compute node makes changes to its instantiation of the application environment. Applicants respectfully assert that the Specification describes that the union block device is a driver that serves to interface between an OS file system in a compute node and the first storage unit and its respective second storage unit (Figure 2; Paragraphs 24-30). Applicants respectfully assert that Paragraphs 28 and 29 state that when the instantiation of the application environment has been changed the union block device will do one of two things: create a new leaf image (thereby replacing the obsolete leaf image) or by modifying the existing leaf image. As the union block device is a driver serving as an interface between the first and second storage devices and the file system of the compute node, Applicants respectfully assert that one skilled in the art would understand how a low-level driver may create or modify a leaf image comprising blocks of data stored in the second storage unit.

Claim Rejections – 35 U.S.C. §101

Claims 23-27 are rejected under 35 U.S.C. 101 because the claimed invention is allegedly directed to non-statutory subject matter. Applicants have amended Claims 24-27 as appropriate. Applicants respectfully request reconsideration and withdrawal of the rejections under 35 U.S.C. §101.

Claim Rejections – 35 U.S.C. §112

Claims 4, 5, 14, 16, and 17 are rejected under 35 U.S.C. 112, second paragraph, as being allegedly indefinite for failing to particularly point out and distinctly claim the subject matter which the application regards as the invention. In particular, and as discussed above, the rejection alleges that the above claims all incorporate "union block devices" with insufficient basis within the claims to define their character. Applicants respectfully disagree. For at least the same or similar reason as discussed above, Applicants respectfully assert that the union block device, as claimed in Claims 4, 5, 14, 16, and 17 would be understood by one skilled in the art to be low-level drivers that are implemented by and for the compute nodes to serve as an interface between the storage units and the file system of each respective compute node. Applicants respectfully assert that one skilled in the art would understand that a union block device is implemented by and for a compute node, rather than existing as a separate "device" requiring a separate processor. Further, Applicants respectfully assert that one skilled in the art would understand that the union block device as a driver serving as an interface between the first and second storage devices and the file system of the compute node, can create or modify a leaf image comprising blocks of data stored in the second storage unit as an instantiation of an operating system is changed by the compute node. Accordingly, Applicants respectfully request reconsideration and withdrawal of the rejections under 35 U.S.C. §112.

Claim Rejections – 35 U.S.C. §102

Claims 14-27 are rejected under 35 U.S.C. 102(b) as being allegedly anticipated by Chang (US 2003/0126242), hereinafter "Chang."

As is well established, anticipation requires the presence of a single prior art reference to disclose each and every element of the claimed invention, arranged as in the claim. There must be no difference between the claimed invention and the reference disclosure, as viewed by a person of ordinary skill in the field of the invention. Scripps Clinic & Research Found. v. Genentech Inc., 18 USPQ 2d 1001, 1010 (Fed. Cir. 1991).

Applicants contend that Chang fails to teach all of the claimed elements of Independent Claims 14, 19, and 23.

Claim 14:

Amended Claim 14 recites in part:

> a plurality of second storage units for storing leaf images of respective compute nodes, said leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes; and
>
> a plurality of union block devices corresponding to said compute nodes, said union block devices for interfacing between said compute nodes and said first and second storage units to distribute said file systems to said compute nodes, wherein said union block devices create said file systems by merging the blocks of said root image stored on the first storage unit with the blocks of respective leaf images stored on respective second storage units, and wherein further at least one of said compute nodes is configurable to index said root image and provide the indexing results to another of said compute nodes (emphasis added).

Applicants respectfully assert that Chang does not teach or suggest "leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes," as claimed in Claim 14.

As understood by Applicants, the rejection is reading the client image copy 74, 76, 78 of Chang on Applicants' claimed "leaf image," as claimed in Claim 14. Applicants respectfully disagree.

As understood by Applicants, Chang discloses a client image copy that is created using a snapshot of a base boot image with a virtual copy or reverse snapshot of the base image stored for each client device, wherein each client image copy includes both a base boot image and information specific to the client (Abstract). In particular, Chang teaches that each client image copy includes both common operating system (OS) and application blocks for storing the base boot image, and client-specific blocks for storing information specific to the particular client device (Paragraph 0012; and Figures 1 and 2). Chang discloses in Figure 2 that each client image copy includes common OS and application blocks 80 containing the reverse snapshot of the base image as well as client specific blocks 82 for storing client specific files (Paragraphs 0030-0034).

To the extent that Chang discloses client image copies that include client-specific blocks for storing information specific to the particular client device, Chang also

discloses that the client image copies also include physically common operating system (OS) and application blocks for storing the base boot image (Paragraph 0012; and Figures 1 and 2). In other words, where Applicants' claim "leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes," as claimed in Claim 14, Chang discloses client image copies that comprise common OS and Application blocks and client specific blocks. Applicants respectfully assert that the client image copy of Chang is not a leaf image, as claimed in Claim 14, because the client image copy of Chang includes blocks of the root image that are unchanged by a compute node (e.g., the common OS and application blocks for storing the base boot image).

Therefore, Applicants respectfully assert that embodiments as recited by Claim 14 are not rendered anticipated by Chang. Accordingly, Applicants respectfully assert that dependent Claims 15-18 are patentable by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

Claim 19:

Independent Claim 19 recites features similar to that of independent Claim 14 and is therefore patentable for at least the same or similar reasons as recited above. Accordingly, Applicants respectfully assert that dependent Claims 20-22 are patentable

by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

Claim 23:

Independent Claim 23 recites features similar to that of independent Claim 14 and is therefore patentable for at least the same or similar reasons as recited above. Accordingly, Applicants respectfully assert that dependent Claims 24-27 are patentable by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

For the above reasons, Applicants request reconsideration and withdrawal of the rejections under 35 U.S.C. §102.

Claim Rejections – 35 U.S.C. §103

Claims 1-13 are rejected under 35 U.S.C. 103(a) as being allegedly unpatentable over Chang, in view of Kobayashi et al. (US 6,101,576), hereinafter "Kobayashi." Applicants respectfully assert that Claims 1-13 are patentable over the cited combination in view of the following.

Applicants respectfully point out that the Examiner has the burden of establishing a prima facie case of obviousness. To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in

the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim features. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on Applicant's disclosure. In re Vaeck, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). See MPEP 2100-126. Specifically, "all words in a claim must be considered when judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d., 1382 (CCPA 1970).

Moreover, in response to the recent U.S. Supreme Court decision in *KSR Int'l Co. v. Teleflex, Inc*. (U.S. 2007), new guidelines were set forth for examining obviousness under 35 U.S.C. 103. The U.S. Supreme Court reaffirmed the *Graham* factors and, while not totally rejecting the "teachings, suggestion, or motivation" test, the Court appears to now require higher scrutiny on the part of the U.S. Patent & Trademark Office. In accordance with the recently submitted guidelines, it is "now necessary to identify the reason" why a person of ordinary skill in the art would have combined the elements of cited references, or at least describe the pertinence of the elements set forth in the cited disclosure, in the manner presently claimed.

Applicants respectfully assert the combination of Chang and Kobayashi fails to teach or suggest all the claimed elements of Claims 1-13 in view of the following rationale.

Claim 1:

Amended Claim 1 recites in part:

> a plurality of second storage units for storing leaf images of respective compute nodes, said leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes (emphasis added).

Applicants respectfully assert the combination of Chang and Kobayashi does not teach or suggest "leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes," as claimed in Claim 1.

For at least the same or similar reasons as recited above with regards to Claim 14, Applicants respectfully assert that Chang does not teach or suggest leaf images comprising only changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes, as claimed in Claim 1. Applicants respectfully assert the deficiencies of Chang are not remedied by Kobayashi.

Therefore, Applicants respectfully assert that Claim 1 is not rendered obvious by the combination of Chang and Kobayashi. Accordingly, Applicants respectfully assert

that dependent Claims 2-6 are patentable by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

Claim 7:

Independent Claim 7 recites features similar to that of independent Claim 1 and is therefore patentable for at least the same or similar reasons as recited above. Accordingly, Applicants respectfully assert that dependent Claims 8-13 are patentable by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

For the above reasons, Applicants request reconsideration and withdrawal of the rejections under 35 U.S.C. §103.

CONCLUSION

In light of the above listed remarks, reconsideration of rejected Claims is requested. Based on the arguments presented above, it is respectfully submitted that Claims 1-27 overcome the rejections of record and, therefore, allowance of Claims 1-27 is earnestly solicited.

Please charge any additional fees that may be required to maintain pendency of the present application, or apply any credits to our PTO deposit account number: 50-4160.

Respectfully submitted,

MURABITO, HAO & BARNES LLP

Dated: August 25, 2010

/Jeffrey A. Lehman/
Jeffrey A. Lehman
Registration No. 65,494

MURABITO, HAO & BARNES LLP
Two North Market Street
Third Floor
San Jose, California 95113

(408) 938-9060 Voice
(408) 938-9069 Facsimile

REMARKS

Claims 1-27 are pending. Claims 5, 8-18, 20-22, and 23-27 are amended. No new matter has been added as a result of these amendments.

Claim Objections

Claims 5, 8-13, 15-18, 20-22, and 24-27 are objected to because of typographical errors. Claims 5, 8-13, 15-18, 20-22, and 24-27 have been amended as appropriate.

Objection to the Specification

Paragraph 22 of the Specification is objected to as it calls out "dashed line 105," while the dashed line in Figure 1 is labeled "106." Paragraph 22 of the Specification has been amended as appropriate.

The Specification is further objected to as failing to provide proper antecedent basis for the claimed subject matter. In particular, the rejection contends that Claims 4, 5, 14, 16, and 17 all allegedly incorporate features called "union block devices" with insufficient basis within the claims to define their character. Applicants respectfully disagree. Applicants respectfully assert that Figure 2 illustrates an exemplary block-level distributed application environment that can contain both physical and virtual elements (Paragraph 26). As illustrated in Figure 2 and discussed in Paragraphs 12 and 22-29, a "union block device" is a driver implemented by and for each respective compute node. As such each union block device driver provides an interface between the first and second storage devices and the file system of its compute node, providing the compute

node with an instantiation of the application environment by accessing the compute node's leaf image containing portions of the application environment that the compute node has changed as well as locating portions of the application environment that have not been changed by the compute node (Paragraphs 27-28). In one exemplary embodiment, the portions that have not been changed by the compute node can be found in the root image (Paragraph 28). Applicants respectfully assert that one skilled in the art would understand the nature and function of such a union block device driver, as described in Paragraphs 12 and 22-29. Applicants respectfully assert that drivers serving as interfaces between operating system file systems for a computer system and associated storage devices are well known in the art. Further, one skilled in the art would understand how a driver may serve as an interface between storage devices and a file system of a computer system.

As understood by Applicants, the rejection contends that Figure 2 illustrates a union block device implemented as a stand-alone "device" that would need its own separate processor. Applicants respectfully disagree. As discussed above, Applicants respectfully assert that Figure 2 illustrates an exemplary block-level distributed application environment that can contain both physical and virtual elements (Paragraph 26). As illustrated in Figure 2 and discussed in Paragraphs 12 and 22-29, a "union block device" is therefore a driver implemented by and for a particular compute node. In other words, union block devices are not "devices" as understood by the rejection, requiring their own separate processor to run the driver. Rather, union block devices are drivers implemented on and for each compute node.

As understood by Applicants, the rejection further questions how a union block device can create a new leaf image on a respective second storage union when the respective compute node makes changes to its instantiation of the application environment. Applicants respectfully assert that the Specification describes that the union block device is a driver that serves to interface between an OS file system in a compute node and the first storage unit and its respective second storage unit (Figure 2; Paragraphs 24-30). Applicants respectfully assert that Paragraphs 28 and 29 state that when the instantiation of the application environment has been changed the union block device will do one of two things: create a new leaf image (thereby replacing the obsolete leaf image) or by modifying the existing leaf image. As the union block device is a driver serving as an interface between the first and second storage devices and the file system of the compute node, Applicants respectfully assert that one skilled in the art would understand how a low-level driver may create or modify a leaf image comprising blocks of data stored in the second storage unit.

Claim Rejections – 35 U.S.C. §101

Claims 23-27 are rejected under 35 U.S.C. 101 because the claimed invention is allegedly directed to non-statutory subject matter. Applicants have amended Claims 24-27 as appropriate. Applicants respectfully request reconsideration and withdrawal of the rejections under 35 U.S.C. §101.

Claim Rejections – 35 U.S.C. §112

Claims 4, 5, 14, 16, and 17 are rejected under 35 U.S.C. 112, second paragraph, as being allegedly indefinite for failing to particularly point out and distinctly claim the subject matter which the application regards as the invention. In particular, and as discussed above, the rejection alleges that the above claims all incorporate "union block devices" with insufficient basis within the claims to define their character. Applicants respectfully disagree. For at least the same or similar reason as discussed above, Applicants respectfully assert that the union block device, as claimed in Claims 4, 5, 14, 16, and 17 would be understood by one skilled in the art to be low-level drivers that are implemented by and for the compute nodes to serve as an interface between the storage units and the file system of each respective compute node. Applicants respectfully assert that one skilled in the art would understand that a union block device is implemented by and for a compute node, rather than existing as a separate "device" requiring a separate processor. Further, Applicants respectfully assert that one skilled in the art would understand that the union block device as a driver serving as an interface between the first and second storage devices and the file system of the compute node, can create or modify a leaf image comprising blocks of data stored in the second storage unit as an instantiation of an operating system is changed by the compute node. Accordingly, Applicants respectfully request reconsideration and withdrawal of the rejections under 35 U.S.C. §112.

Claim Rejections – 35 U.S.C. §102

Claims 14-27 are rejected under 35 U.S.C. 102(b) as being allegedly anticipated by Chang (US 2003/0126242), hereinafter "Chang."

As is well established, anticipation requires the presence of a single prior art reference to disclose each and every element of the claimed invention, arranged as in the claim. There must be no difference between the claimed invention and the reference disclosure, as viewed by a person of ordinary skill in the field of the invention. Scripps Clinic & Research Found. v. Genentech Inc., 18 USPQ 2d 1001, 1010 (Fed. Cir. 1991).

Applicants contend that Chang fails to teach all of the claimed elements of Independent Claims 14, 19, and 23.

Claim 14:

Amended Claim 14 recites in part:

> a plurality of second storage units for storing leaf images of respective compute nodes, said leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes; and
>
> a plurality of union block devices corresponding to said compute nodes, said union block devices for interfacing between said compute nodes and said first and second storage units to distribute said file systems to said compute nodes, wherein said union block devices create said file systems by merging the blocks of said root image stored on the first storage unit with the blocks of respective leaf images stored on respective second storage units, and wherein further at least one of said compute nodes is configurable to index said root image and provide the indexing results to another of said compute nodes (emphasis added).

Applicants respectfully assert that Chang does not teach or suggest "leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes," as claimed in Claim 14.

As understood by Applicants, the rejection is reading the client image copy 74, 76, 78 of Chang on Applicants' claimed "leaf image," as claimed in Claim 14. Applicants respectfully disagree.

As understood by Applicants, Chang discloses a client image copy that is created using a snapshot of a base boot image with a virtual copy or reverse snapshot of the base image stored for each client device, wherein each client image copy includes both a base boot image and information specific to the client (Abstract). In particular, Chang teaches that each client image copy includes both common operating system (OS) and application blocks for storing the base boot image, and client-specific blocks for storing information specific to the particular client device (Paragraph 0012; and Figures 1 and 2). Chang discloses in Figure 2 that each client image copy includes common OS and application blocks 80 containing the reverse snapshot of the base image as well as client specific blocks 82 for storing client specific files (Paragraphs 0030-0034).

To the extent that Chang discloses client image copies that include client-specific blocks for storing information specific to the particular client device, Chang also

discloses that the client image copies also include physically common operating system (OS) and application blocks for storing the base boot image (Paragraph 0012; and Figures 1 and 2). In other words, where Applicants' claim "leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes," as claimed in Claim 14, Chang discloses client image copies that comprise common OS and Application blocks and client specific blocks. Applicants respectfully assert that the client image copy of Chang is not a leaf image, as claimed in Claim 14, because the client image copy of Chang includes blocks of the root image that are unchanged by a compute node (e.g., the common OS and application blocks for storing the base boot image).

Therefore, Applicants respectfully assert that embodiments as recited by Claim 14 are not rendered anticipated by Chang. Accordingly, Applicants respectfully assert that dependent Claims 15-18 are patentable by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

Claim 19:

Independent Claim 19 recites features similar to that of independent Claim 14 and is therefore patentable for at least the same or similar reasons as recited above. Accordingly, Applicants respectfully assert that dependent Claims 20-22 are patentable

by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

Claim 23:

Independent Claim 23 recites features similar to that of independent Claim 14 and is therefore patentable for at least the same or similar reasons as recited above. Accordingly, Applicants respectfully assert that dependent Claims 24-27 are patentable by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

For the above reasons, Applicants request reconsideration and withdrawal of the rejections under 35 U.S.C. §102.

Claim Rejections – 35 U.S.C. §103

Claims 1-13 are rejected under 35 U.S.C. 103(a) as being allegedly unpatentable over Chang, in view of Kobayashi et al. (US 6,101,576), hereinafter "Kobayashi." Applicants respectfully assert that Claims 1-13 are patentable over the cited combination in view of the following.

Applicants respectfully point out that the Examiner has the burden of establishing a prima facie case of obviousness. To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in

the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim features. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on Applicant's disclosure. In re Vaeck, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). See MPEP 2100-126. Specifically, "all words in a claim must be considered when judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d., 1382 (CCPA 1970).

Moreover, in response to the recent U.S. Supreme Court decision in *KSR Int'l Co. v. Teleflex, Inc*. (U.S. 2007), new guidelines were set forth for examining obviousness under 35 U.S.C. 103. The U.S. Supreme Court reaffirmed the *Graham* factors and, while not totally rejecting the "teachings, suggestion, or motivation" test, the Court appears to now require higher scrutiny on the part of the U.S. Patent & Trademark Office. In accordance with the recently submitted guidelines, it is "now necessary to identify the reason" why a person of ordinary skill in the art would have combined the elements of cited references, or at least describe the pertinence of the elements set forth in the cited disclosure, in the manner presently claimed.

Applicants respectfully assert the combination of Chang and Kobayashi fails to teach or suggest all the claimed elements of Claims 1-13 in view of the following rationale.

Claim 1:

Amended Claim 1 recites in part:

> a plurality of second storage units for storing leaf images of respective compute nodes, said leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes (emphasis added).

Applicants respectfully assert the combination of Chang and Kobayashi does not teach or suggest "leaf images comprising only additional data blocks not previously contained in said root image and changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes," as claimed in Claim 1.

For at least the same or similar reasons as recited above with regards to Claim 14, Applicants respectfully assert that Chang does not teach or suggest leaf images comprising only changes made by respective compute nodes to the blocks of said root image, wherein said leaf images of respective compute nodes do not include blocks of said root image that are unchanged by respective compute nodes, as claimed in Claim 1. Applicants respectfully assert the deficiencies of Chang are not remedied by Kobayashi.

Therefore, Applicants respectfully assert that Claim 1 is not rendered obvious by the combination of Chang and Kobayashi. Accordingly, Applicants respectfully assert

that dependent Claims 2-6 are patentable by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

Claim 7:

Independent Claim 7 recites features similar to that of independent Claim 1 and is therefore patentable for at least the same or similar reasons as recited above. Accordingly, Applicants respectfully assert that dependent Claims 8-13 are patentable by virtue of their dependency on an allowable base claim, as well as for their additional recited patentable features.

For the above reasons, Applicants request reconsideration and withdrawal of the rejections under 35 U.S.C. §103.

CONCLUSION

In light of the above listed remarks, reconsideration of rejected Claims is requested. Based on the arguments presented above, it is respectfully submitted that Claims 1-27 overcome the rejections of record and, therefore, allowance of Claims 1-27 is earnestly solicited.

Please charge any additional fees that may be required to maintain pendency of the present application, or apply any credits to our PTO deposit account number: 50-4160.

Respectfully submitted,

MURABITO, HAO & BARNES LLP

Dated: 08/31, 2010

/Jeffrey A. Lehman/
Jeffrey A. Lehman
Registration No. 65,494

MURABITO, HAO & BARNES LLP
Two North Market Street
Third Floor
San Jose, California 95113

(408) 938-9060 Voice
(408) 938-9069 Facsimile

I hereby certify that this correspondence is being electronically transmitted to the USPTO via EFS on the date shown below:
August 31, 2010
/Shannon Warren/
Shannon Warren